PROGRAMMABLE LOGIC CONTROLLER[*]

ANTONIO SORIN TASU

Department of Physics, Ovidius University, Constanța, 900527, Romania

Received December 21, 2004

The purpose of this Programmable Logic Controller (PLC) is to command 16 relays in different sequences programmed by user. This device is appliable for controlling and monitoring industrial processes and may be implemented in multiple domains, from auto industry and home control applications to industrial instruments, remote sensors and security devices. It's also ideal for battery powered devices because it's low power consumption. EEPROM memory make easier the implementation of the microcontroller on devices where permanent memory is needed to store different parameters (codes for transmitters,motor speed,receiver frequencies). The controller ca be used in small industry in controling and monitoring industrial processes. Depending of the programmed software, the device can *i*) command a packing line *ii*) count the products from a industrial process line and display the result on any kind of display, and *iii*) communicate with dedicated PLC's using digital level shifters.

Key words: Programmable Logic Controller, EEPROM.

1. INTRODUCTION

PLCs (programmable logic controllers) are the control hubs for a wide variety of automated systems and processes. They contain multiple inputs and outputs that use transistors and other circuitry to simulate switches and relays to control equipment. They are programmable via software interfaced via standard computer interfaces and proprietary languages and network options. The development of low cost computer has brought the most recent revolution, the Programmable Logic Controller (PLC). The advent of the PLC began in the 1970s, and has become the most common choice for manufacturing controls. PLCs have been gaining popularity on the factory floor and will probably remain predominant for some time to come. Most of this is because of the advantages they offer. Are flexible and can be reapplied to control other systems quickly and easily, computational abilities allow more sophisticated control, reliable components make these likely to operate for years before failure.

[*] Paper presented at the 5th International Balkan Workshop on Applied Physics, 5–7 July 2004, Constanța, Romania.

The Programmable Logic Controller presented here is built arround a PIC 16F877 microcontroller made by Microchip Company [6]. This processor with RISC (Reduced Instruction Set Controller) architecture is a part of “midrange” controllers and works on 8 bits. It has 5 external ports which can be programmed as digital inputs or outputs. Totally, 33 pins are available which can be used in interaction with exterior world. Current drivers Q1-Q16 are also a part of the schematic witch drive the control LEDs to visualise the outputs states. This microcontroller can be programmed through a hardware programmer connected to the PC’s serial port; this programmer makes digital level conversions from serial port levels to levels that microcontroller needs. Two programming languages can be used to generate a *.hex file which is loaded in microcontroller’s flash memory: -assembler-provided by the manufacturer (35 instructions), -JAL High Level Language-designed by the Sweedish programmer Wouter van Ooijen [1] and Romanian engineer Vasile Surducan [2]. For transferring data from PC to microcontroller, Icprog software is used.

2. EXPERIMENTAL

This controller can command up to 16 relays on two 8 bits external ports, these outputs take values depending of the 6 inputs (F1-F6) states. Ports X and Y states are displayed on a 16 characters, 2 lines matricial LCD display.

The PIC16F877 has a 13-bit program counter capable of addressing an 8K x 14 program memory space, as shown in Fig. 3.

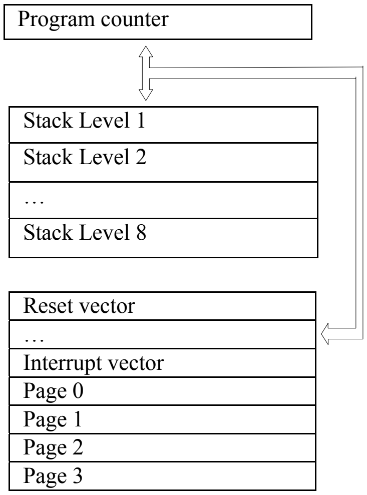

Fig. 3 – Memory structure.

The data memory is partitioned into multiple banks which contain the General Purpose Registers and the Special Function Registers. Bits RP1 and RP0 are the bank select bits.

Each bank extends up to 7Fh (128 bytes). The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers, implemented as static RAM. All implemented banks contain Special Function Registers. Some frequently used Special Function Registers from one bank may be mirrored in another bank for code reduction and quicker access.

The maximum frequency that can be reached on microcontroller's pin is given by this relation:

$$F_{out}=F_{osc}/4$$

where F_{out} is pin frequency and F_{osc} is timing oscillator frequency. In this project a 20 MHz quartz is used, so the pin frequency is 5 MHz. Microcontroller outputs are open-drain and doesn't support a greater current than 20 mA. For this reason Q1-Q16 drivers have been used. The drivers are working in switching mode (saturation-blocking), the current through the LEDs is limited by R2 resistors. Also a realtime clock can be programmed to establish user programmable delays anywhere in the range between miliseconds and hours.

3. DESIGN

The basic idea was to design a PLC with performances comparable to dedicated industrial PLC's designed by big companies like Siemens, Telemecanique, etc. The target was to obtain a lower price for the final device. The block diagram of the PLC, presented in Fig. 1, consists of a microcontroller with flash memory, an user controlled keyboard, an alphanumeric display to control the state of the physical I/O lines and the internal state of different variables that can be used, power drivers for motors, steppers, displays, etc. and different peripherals that can be connected to the device.

I have used a midrange microcontroller from Microchip Corp. which is less expensive than Intel microcontrollers used in dedicated PLC's but with comparable performances. They have the same numbers of I/O lines, are working at the same clock frequencies, on both devices the flash memory can be programmed 10.000 times.

As integrated peripherals, I have used an HD44780 alphanumeric display which is less expensive than a graphic display and it is easier to program. The power supply module (see Fig. 2) is built in a classic way, a 5V regulator is used to asign the right voltage and a 100nF capacitor to cut the spikes that can appear and to keep stable the whole system. The resistors R1 was used to pull down the inputs.

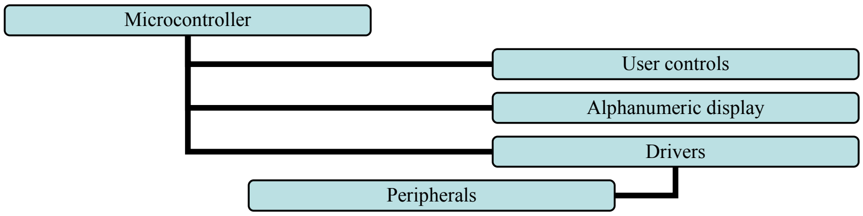

Fig. 1 – Hardware block diagram.

Example: here is shown a simple conversion from decimal to binary base. X is a byte type variable which is the physical 8 bits external port X. For quick user programming, on the matricial LCD this variable is shown in decimal. User must set this variable to the desired value then press the F1 touch button (F1 is the physical input number 1) and the X port output will take the preprogrammed value. Instead of the F1 button the user can use a photo barrier, a mechanical switch or other peripheral depending of his needs.

PG01-IF F1 THEN PORT X STATE

LCD	X8	X7	X6	X5	X4	X3	X2	X1
X=160	1	0	1	0	0	0	0	0

4. CONCLUSIONS

I have designed, implemented, tested and demonstrated the functionality of a complete Programmable Logic Controller built around a PIC 16F877 microcontroller from Microchip Corp. I showed that the PLC is able to control peripheral devices such as relays, any kind of displays, motors, steppers, etc. The maximum values for currents and voltages, as indicated by the manufacturer were reached, with little differences on internal open-drain output current drivers. It is also possible to overclock the device to raise the working speed by up to 10%. This PLC can be used to control small scale manufacturing processes, to count various products in a industrial process, to build large public clocks, commercial displays, etc. The Programmable Logic Controller presented here costs about 50 euro and the average price for the PLC's on the market is between 700-1000 euro.

Acknowledgments. The author thanks Vasile Surducan from Cluj-Napoca, for useful discussions regarding midrange microcontrollers from Microchip Corp.

REFERENCES

1. Wouter van Ooijen - www.voti.nl/jal/.
2. Vasile Surducan, Microcontrolere pentru toti, Editura Risoprint 2003 (in Romanian).
3. Mihai Dinca, Manualul Studentului, University Press 2002 (in Romanian).
4. Barry Wilkinson, The Essence of Digital Design, Prentice Hall Europe, a Pearson Education Company-1997.
5. John F. Wakerly, Digital Design:Principles and practices-Prentice Hall Inc., a Pearson Education Company, 2000.
6. Microchip Corporation-www.microchip.com.